CLAIMS

What is claimed is:

1. An accelerated processor for use in massive data manipulations specific to an application comprising:

a workstation having a general purpose processor and a coprocessor connection;

an application specific coprocessor system at said connection;

said coprocessor system having programming code which is a assembled as instructions for said specific application in combination with accelerator environment specific requirements, independently provided.

2. An application specific coprocessor system for use with a processor for use in massive data manipulations specific to an application and adapted for attachment to a workstation having a general purpose processor, said coprocessor system having programming code which is assembled as instructions for said specific application in combination with accelerator environment specific requirements, independently provided.

3. The coprocessor of claims 1 or 2 wherein:

said environment specific instructions are accessed by a compiler in response to user input in an application specific form.

4. The coprocessor of claim 3 wherein said compiler comprises one or more of: user interface to permit an application trained non circuit design trained user to enter instructions to achieve accelerated performance, means to create

an internal representation reflecting the operational characteristics of a coprocessor corresponding to application specific accelerated processing needs, means for identifying bit demands for the application specific coprocessor acceleration function, mapper means for identifying resources available and needed for the coprocessor to provide application specific accelerated processing, balancing means for identifying the step by step hardware needs of the coprocessor for the application specific acceleration.

5. The coprocessor of claim 4 wherein said mapper means accepts as input domain-specific policy information, estimates of the amount of logic needed for each processing element, and hardware context information that states what amounts of each logic resource exist on a given coprocessor to enable the largest possible number of processing elements said coprocessor can support.

6. The coprocessor of claim 5 wherein said balancing means analyzes the processing speed of said coprocessor at each step and allocates parallel hardware in proportion to a speed requirement.

7. The coprocessor of claim 3, 4, 5 or 6 wherein said compiler further includes one or more of prerecorded information; reflecting the programming requirements for a general area of applications; programming content which reflects application requirements and hardware characteristics; and coprocessor specific hardware availability.

8. A method for programming an accelerating coprocessor comprising the steps of:

accessing data reflective of programming requirements for a general area of applications.

9. The method for programming an accelerating coprocessor of claim 8 comprising the steps of:

accessing data reflective of programming content which reflects application requirements and hardware characteristics.

10. The method for programming an accelerating coprocessor of claim 8 or 9 comprising the steps of:

accessing data reflective of coprocessor specific hardware availability.

11. The method of any of claims 8 through 10 further comprising the steps of:

permitting an application trained non circuit design trained user to enter instructions to achieve accelerated performance.

12. The method of any of claims 8 through 11 further comprising the steps of:

creating an internal representation reflecting the operational characteristics of a coprocessor corresponding to application specific accelerated processing needs.

13. The method of any of claims 8 through 12 further comprising the steps of:

identifying bit demands for the application specific coprocessor acceleration function, means for identifying correlating available and needed for the coprocessor to provide application specific accelerated processing.

14. The method of any of claims 8 through 13 further comprising the steps of:

identifying the step by step hardware needs of the coprocessor for the application specific acceleration.

15. A method of compiling data for programming an accelerating coprocessor comprising the steps of:

permitting an application trained non circuit design trained user to enter instructions to achieve accelerated performance

16. A method of compiling data for programming an accelerating coprocessor comprising the steps of:

creating an internal representation reflecting the operational characteristics of a coprocessor corresponding to application specific accelerated processing needs, means for identifying bit demands for the application specific coprocessor acceleration function.

17. A method of compiling data for programming an accelerating coprocessor comprising the steps of:

identifying resources available and needed for the coprocessor to provide application specific accelerated processing.

18. The method of claim 17 wherein said identifying step further includes the step of accepting as input domain-specific policy information, estimates of an amount of logic needed for each processing element, and hardware context information that states what amounts of each logic resource exist in a given coprocessor and providing a design maximizing a number of processing elements that the coprocessor can support.

19. A method of compiling data for programming an accelerating coprocessor comprising the steps of:

identifying the step by step hardware needs of the coprocessor for the application specific acceleration.

20. The method of claim 19 wherein said identifying step includes analyzing the processing speed at each step and allocating parallel hardware in proportion to a processing speed requirement.

21. A compiler for programming an accelerating coprocessor comprising:

means for accessing data reflective of programming requirements for a general area of applications.

22. The compiler for programming an accelerating coprocessor of claim 21 further comprising:

means for accessing data reflective of programming content which reflects application requirements and hardware characteristics.

23. The compiler for programming an accelerating coprocessor of claim 21 or 22 further comprising:

means for accessing data reflective of coprocessor specific hardware availability.

24. The compiler of any of claims 21 through 23 further comprising:

means for permitting an application trained non circuit design trained user to enter instructions to achieve accelerated performance.

25. The compiler of any of claims 21 through 24 further comprising:

means for creating an internal representation reflecting the operational characteristics of a coprocessor corresponding to application specific accelerated processing needs.

26. The compiler of any of claims 21 through 25 further comprising:

means for identifying bit demands for the application specific coprocessor acceleration function.

27. The compiler of any of claims 21 through 26 further comprising means for identifying resources available and needed for the coprocessor to provide application specific accelerated processing.

28. The compiler of any of claims 21 through 27 further comprising:

means for identifying the step by step hardware needs of the coprocessor for the application specific acceleration.

29. A compiler for data for programming an accelerating coprocessor comprising:

means for permitting an application trained non circuit design trained user to enter instructions to achieve accelerated performance

30. A compiler for data for programming an accelerating coprocessor comprising:

means for creating an internal representation reflecting the operational characteristics of a coprocessor corresponding

to application specific accelerated processing needs, means for identifying bit demands for the application specific coprocessor acceleration function.

31. A compiler for data for programming an accelerating coprocessor comprising:

means for identifying resources available and needed for the coprocessor to provide application specific accelerated processing.

32. The compiler of claim 31 wherein said identifying means further includes means for accepting as input domain-specific policy information, estimates of an amount of logic needed for each processing element, and hardware context information that states what amounts of each logic resource exist in a given coprocessor and providing a design maximizing a number of processing elements that the coprocessor can support.

33. A compiler for data for programming an accelerating coprocessor comprising:

means for identifying the step by step hardware needs of the coprocessor for the application specific acceleration.

34. The compiler of claim 33 wherein said identifying means includes means for analyzing the processing speed at each step and allocating parallel hardware in proportion to a processing speed requirement.